

A Fuzzy-based Adaptive Genetic Algorithm and Its Case Study in Chemical Engineering^{*}

YANG Chuanxin (杨传鑫) and YAN Xuefeng (颜学峰)^{**}

Key Laboratory of Advanced Control and Optimization for Chemical Processes of Ministry of Education, East China University of Science and Technology, Shanghai 200237, China

Abstract Considering that the performance of a genetic algorithm (GA) is affected by many factors and their relationships are complex and hard to be described, a novel fuzzy-based adaptive genetic algorithm (FAGA) combined a new artificial immune system with fuzzy system theory is proposed due to the fact fuzzy theory can describe high complex problems. In FAGA, immune theory is used to improve the performance of selection operation. And, crossover probability and mutation probability are adjusted dynamically by fuzzy inferences, which are developed according to the heuristic fuzzy relationship between algorithm performances and control parameters. The experiments show that FAGA can efficiently overcome shortcomings of GA, *i.e.*, premature and slow, and obtain better results than two typical fuzzy GAs. Finally, FAGA was used for the parameters estimation of reaction kinetics model and the satisfactory result was obtained.

Keywords fuzzy logic controller, genetic algorithm, artificial immune system, reaction kinetics model

1 INTRODUCTION

Genetic algorithm (GA) is based on Darwin's theory of evolution and regards optimization as an evolution process [1]. Due to its many virtues on searching optimum and the recent theory progress, many industrial applications, such as function optimizations, self-adaptive control systems and so on, have been solved *via* the aid of GA. But GA [2] has two main shortcomings, *i.e.* premature convergence and low searching efficiency at the anaphase.

GA has three control parameters: population size N , crossover rate p_c and mutation rate p_m . The effectiveness, efficiency and robustness of GA are sensitive to the settings of control parameters with various degrees [3]. In particular, the mutation operation and the crossover operation play an important role in the performance of GA. In the original GA, two control parameters are constant during all evolution process. Two optimal parameter values are difficult to be obtained and it conflicts between constant parameters and random and dynamic searching process of GA. As a solution, self-adaptation was incorporated to adjust control parameters, in which the two control parameters p_c and p_m are not required to be pre-specified. With population evolution, the parameter values are self-adapted according to the learning experience [4]. The choice of N is typically regarded as a 'fine tuning parameter' in GA. Thus, self-adaptation is usually applied to p_c and p_m control parameters.

The literatures [5] introduced the self-adaptation methods based on fuzzy logic controller (FLC) to adjust control parameters of GA. A fuzzy adaptive genetic algorithm [1] with variable population size uses

both macroscopical and microscopical control based on heuristic rules to realize population size adaptation, overcome premature convergence and slow convergence speed at later evolution process. In Section 2, two typical kinds of fuzzy adaptive genetic algorithms [6, 7] are introduced and employed in the paper for performance comparison.

In this paper, a fuzzy-based adaptive genetic algorithm (FAGA) combined with new artificial immune system and fuzzy system theory is proposed. In FAGA, immune theory is used to improve the selection operation of GA. And, crossover and mutation probabilities are adjusted dynamically by fuzzy inferences based on the heuristic fuzzy relationship between algorithm performances and control parameters. In FAGA, the convergence is enhanced and the diversity of population is maintained at the same time. Experimental results show that FAGA effectively overcomes slow and premature, and the whole performance is better than two mentioned fuzzy adaptive genetic algorithms [6, 7].

2 TWO TYPICAL FUZZY GENETIC ALGORITHMS

2.1 Fuzzy logic control

A fuzzy logic controller [8] (FLC) comprises the following four components [9]: a fuzzification interface, a rule base, a decision-making logic and a defuzzification interface. In the fuzzification interface, the measured crisp values of input variables are converted into suitable linguistic values viewed as labels of fuzzy sets. The rule base characterizes the control

Received 2010-05-05, accepted 2010-10-29.

^{*} Supported by the National Natural Science Foundation of China (20776042), the National High Technology Research and Development Program of China (2007AA04Z164), the Doctoral Fund of Ministry of Education of China (20090074110005), the Program for New Century Excellent Talents in University (NCET-09-0346), the "Shu Guang" Project (095G29), and Shanghai Leading Academic Discipline Project (B504).

^{**} To whom correspondence should be addressed. E-mail: xfyang@ecust.edu.cn

goals and control strategies of the domain expert by means of a set of linguistic control rule. The decision-making logic, which emulates human decision-making behavior based on fuzzy concepts, infers fuzzy control actions employing fuzzy implication and the rules of inference in fuzzy. In the defuzzification interface, the inferred fuzzy control actions are then converted into actual non-fuzzy one.

2.2 Method 1 of genetic algorithm based on FLC (FGA 1)

FGA 1, a typical fuzzy adaptive genetic algorithm, is described in the literature in detail [7].

2.2.1 Input 1 of FLC

ES, one input variable of FLC, is determined by the average fitness values of previous generation and contemporary generation [7] is as follows.

$$ES(t+1) = \frac{\frac{1}{N} \sum_{j=1}^N \text{fitness}(t+1, j) - \frac{1}{N} \sum_{j=1}^N \text{fitness}(t, j)}{\frac{1}{N} \sum_{j=1}^N \text{fitness}(t, j)} \quad (1)$$

where $ES(t)$ is evolutionary speed at the t -th generation, N is the number of population and $\text{fitness}(t, i)$ is the fitness value of the i th individual at the t -th generation.

2.2.2 Input 2 of FLC

AD (Average Distance), another input variable of FLC, is the average distance from each individual to the center of the population.

$$AD(t) = \frac{1}{N} \sum_{i=1}^N \sqrt{(x_{p,1} - x_{i,1})^2 + \cdots + (x_{p,n} - x_{i,n})^2} \quad (2)$$

where $x_{p,j} = \frac{1}{N} \sum_{j=1}^N x_{j,i}$ is the average value of the i -th element of individual and $x_{j,i}$ is the value of the i -th element of the j -th individual.

The more AD is, the greater diversity of population is, and *vice versa*. Crossover rule base is shown in Table 1. When ES is lower, crossover rate should be increased to produce the best individual. When AD is lower, crossover rate should be increased to make greater diversity of population. Especially, when both ES and AD are low, it is easy to fall into prematurity and higher crossover rate is needed to produce new individuals.

Table 1 Crossover rule base

ES	AD		
	Low	Medium	High
low	higher	high	medium
medium	high	medium	low
high	medium	low	lower

Mutation rule base is shown in Table 2. When ES is higher, mutation rate should be reduced to protect great individuals. When AD is higher, mutation rate should be reduced to promote the convergence of population. When both ES and AD are low, population almost fall into prematurity and it needs higher mutation rate.

Table 2 Mutation rule base

ES	AD		
	Low	Medium	High
low	higher	medium	low
medium	high	medium	low
high	medium	low	lower

2.3 Method 2 of genetic algorithm based on FLC (FGA 2)

FGA 2, another typical fuzzy adaptive genetic algorithm, is described in the literature in detail [6].

2.3.1 Input 1 of FLC 2

The degree of individual differences S_b [6], one input variable of FLC, is defined as follows.

Define 1 Individual $\mathbf{x}_t^i$ is composed by n genes [6] at the t -th generation,

$$\mathbf{x}_t^i = [x_t^{i(1)}, x_t^{i(2)}, x_t^{i(3)}, \dots, x_t^{i(n)}], \quad i \in \{1, 2, 3, \dots, N\} \quad (3)$$

Average individual at the t -th generation is defined as follows.

$$\bar{\mathbf{x}}_t = [\bar{x}_t^{(1)} \cdots \bar{x}_t^{(d)} \cdots \bar{x}_t^{(n)}], \quad \bar{x}_t^{(d)} = \frac{1}{N} \sum_{i=1}^N x_t^{i(d)} \quad (4)$$

where N is the number of population at the t -th generation.

Define 2 Distance between individual $\mathbf{x}_t^i$ and average individual at the t -th generation is calculated as follows.

$$\partial(\mathbf{x}_t^i, \bar{\mathbf{x}}_t) = \sqrt{\sum_{j=1}^n (x_t^{i(j)} - \bar{x}_t^{(j)})^2} \quad (5)$$

Define 3 The degree of individual differences at the t -th generation is calculated as follows.

$$S_b = \frac{1}{N} \sum_{i=1}^N \partial(\mathbf{x}_t^i, \bar{\mathbf{x}}_t) \quad (6)$$

A significant feature of population falling in a local optimal solution is that the difference between individuals becomes very small. Namely, S_b is small. When the degree of individual differences is big, S_b is big and the diversity of population is great.

2.3.2 Input 2 of FLC 2

Variance E , another input variable of FLC, can be used to forecast the occurrence of premature convergence and is defined as follows.

$$E = \frac{1}{N} \sum_{i=1}^N [f(\mathbf{x}_i) - \bar{f}]^2 \quad (7)$$

where $\bar{f} = \frac{1}{N} \sum_{i=1}^N f(\mathbf{x}_i)$ is the average value of fitness values.

2.3.3 Algorithm

Selection: Selection probability P_s of individual

$\mathbf{x}^i$ is defined as follows: $P_s = \frac{f_i}{\sum_{i=1}^N f_i}$, in which f_i is the

fitness value of individual $\mathbf{x}^i$.

Crossover: the point c is determined randomly.

Parent individual 1:

$$\mathbf{x}^1 = \{x^{1(1)}, x^{1(2)}, \dots, x^{1(c)}, \dots, x^{1(n)}\}$$

Parent individual 2:

$$\mathbf{x}^2 = \{x^{2(1)}, x^{2(2)}, \dots, x^{2(c)}, \dots, x^{2(n)}\}$$

Two offspring individuals are produced as follows.

$$x_n^{1(c)} = x^{1(c)} - \beta(x^{1(c)} - x^{2(c)}),$$

$$x_n^{2(c)} = x^{1(c)} + \beta(x^{1(c)} - x^{2(c)}), \quad \beta \in [0, 1]$$

Offspring individual 1:

$$\mathbf{x}_n^1 = \{x^{1(1)}, x^{1(2)}, \dots, x^{2(c)}, \dots, x^{2(n)}\}$$

Offspring individual 2:

$$\mathbf{x}_n^2 = \{x^{2(1)}, x^{2(2)}, \dots, x^{1(c)}, \dots, x^{1(n)}\}$$

Mutation: Mutation operation is shown as the following formula.

$$\varphi(\mathbf{x}^i, l) = x^{i(l)} + N(0, \sigma)$$

where $\mathbf{x}^i$ is individual, l is the l -th gene and $N(0, \sigma)$ is Gaussian noise whose variance is σ and average value is 0.

3 FUZZY-BASED ADAPTIVE GENETIC ALGORITHM

3.1 Criterion of premature convergence

In order to forecast the occurrence of premature convergence, a new criterion about premature convergence is defined as follows.

$$S = (\overline{f_{t\max}} - \overline{f_{t\min}}) / (f_{t\max} - f_{t\min}) \quad (8)$$

where $f_{t\max}$ and $f_{t\min}$ are the maximum and minimum of fitness value at the t -th generation, respectively, $\overline{f_t} = \frac{1}{N} \sum_{i=1}^N f_t^i$ is the average value of fitness values at the t -th generation, $\overline{f_{t\max}}$ is the average

value of those fitness values which are more than $\overline{f_t}$ at the t -th generation, $\overline{f_{t\min}}$ is the average value of those fitness values which are less than $\overline{f_t}$ at the t -th generation and S indicates the distribution of individuals.

3.2 Selection probability of immune genetic algorithm (IGA)

Because of selection pressure, it is possible to make some excellence individuals occupy the dominant position quickly in the evolution process of GA. The global optimal solution could not be found as a result of lacking new genes. Selection probability of IGA [10] derives the concept of immune and can effectively avoid the premature. In IGA, it regards antigen as objective function, antibody as solution and the affinity between antigen and antibody as the combined intensity of solution. In FAGA, the selection probability of individual is calculated the same as the selection probability [11] of IGA.

Definition 1 In giving antibody population, the concentration of antibody v is defined as follows [12]:

$$c_v = \frac{1}{N} \sum_{w=1}^N A_{v,w} \quad (9)$$

where $A_{v,w} = \begin{cases} 1 & a_{v,w} \geq T \\ 0 & \text{others} \end{cases}$, $a_{v,w}$ is the affinity between antibody v and antibody w , and it expresses similarity of antibodies, N is total number of antibodies in giving antibody population and T is pre-given threshold.

Definition 2 f_v and f_w are the fitness values of antibody v and w . Given the appropriate positive number m , if $|f_v - f_w| < m$, $A_{v,w} = 1$. Otherwise, $A_{v,w} = 0$. The concentration of antibody is shown in Eq. (9).

According to the above definition, the expected reproduction probability of antibody v ($v = 1, 2, \dots, N$) is e_v defined as follows.

$$e_v = \frac{f_v}{c_v} \quad (10)$$

The selection probability p_1 of antibody v ($v = 1, 2, \dots, N$) is defined as follows.

$$p_1 = \frac{e_v}{\sum_{i=1}^N e_i} \quad (11)$$

3.3 Design of FLC

3.3.1 Input variables of FLC

In FAGA, input variable 1 of FLC was defined as follows [13].

$$p = g / g_{\max} \quad (12)$$

where g means that current generation is the g -th

generation and $g_{\max}$ is the total number of generation. In FAGA, input variable 2 of FLC was defined as Eq. (8). When S is very small, the diversity is good. p_c and p_m should be small enough. When S is big enough, p_c and p_m should become big to increase the diversity.

S and p was defined as two input variables of FLC, and p_c and p_m are two output variables of FLC. Principle of FLC is shown in Fig. 1. Membership functions of S , p , crossover probability and mutation probability are all Gauss.

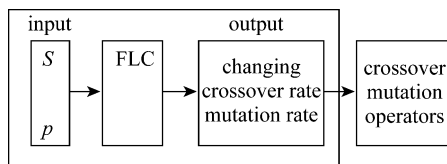


Figure 1 Principle of FLC

3.3.2 Fuzzification of crossover-rate

It was defined that $p_c \in \{\text{lower, low, medium, high, higher, highest}\}$. Crossover-rate may be too low to produce good individuals. Therefore, crossover-rate is between 0.5 and 1. Membership function of p_c is

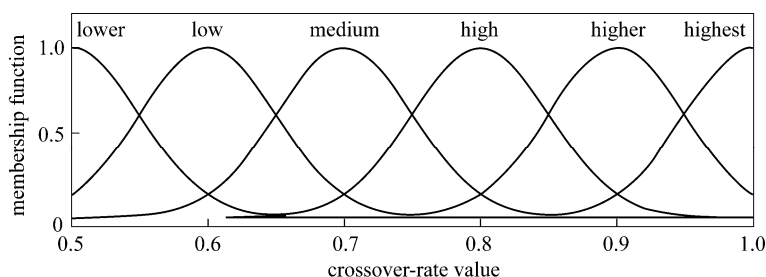


Figure 2 Membership function of p_c

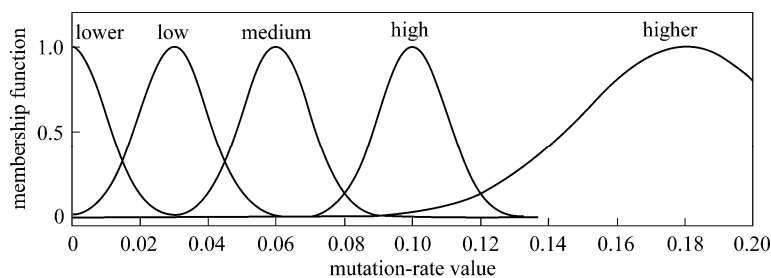


Figure 3 Membership function of p_m

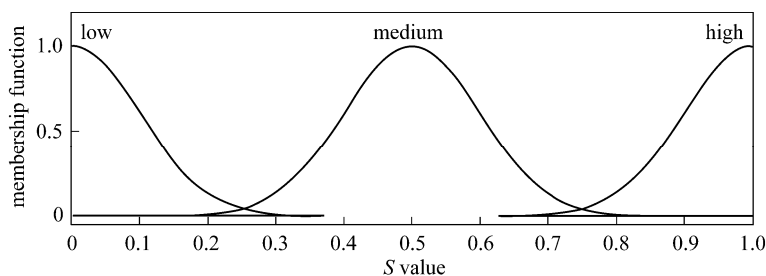


Figure 4 Membership function of S

shown in Fig. 2.

3.3.3 Fuzzification of mutation-rate

It was defined that $p_m \in \{\text{lower, low, medium, high, higher}\}$. Mutation-rate could not be too big, otherwise the genetic algorithm would become random search. Thus, mutation-rate is between 0 and 0.2. Membership function of p_m is shown in Fig. 3.

3.3.4 Fuzzification of S

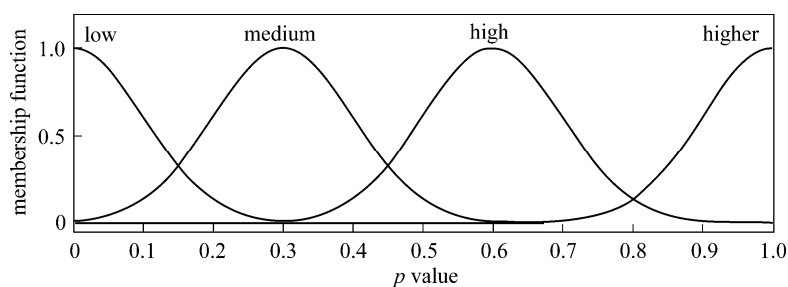
It was defined that $S \in \{\text{low, medium, high}\}$. S is between 0 and 1. Membership function of S is shown in Fig. 4.

3.3.5 Fuzzification of p

It was defined that $p \in \{\text{low, medium, high, higher}\}$. p is between 0 and 1. Membership function of p is shown in Fig. 5.

3.3.6 Fuzzy rules

Fuzzy inference system is based on the rules of knowledge base. Each fuzzy state has a corresponding output fuzzy subset, which could make algorithm more flexible. In this paper we have two inputs. One of them owns three linguistic values and another has four linguistic values. Thus, the controller involves 3×4

Figure 5 Membership function of p

fuzzy control rules which take the form as follows [8]: Rule i : if S is A_i and p is B_i , then p_c is C_i and p_m is D_i ($i=1,2,\dots,H$). Where A_i and B_i are the fuzzy sets of the linguistic variables, C_i and D_i are the corresponding point values of fuzzy singleton and H is the total number of control rules. Crossover rule base is shown in Table 3.

Table 3 Crossover rule base

p	S		
	Low	Medium	High
low	high	low	medium
medium	medium	medium	higher
high	lower	high	higher
higher	high	high	high

When the diversity is good (S is low), p_c should be increased. When p is low, it shows that algorithm is running at prophase. In this time, p_c should be big enough to enhance its search. At the medium time, S often becomes high and it is easy to be premature convergence. In order to produce some new individuals, p_c needs to be higher. And at the last period, S also often becomes high and p_c should become big to search the optimal solution.

Mutation rule base is shown in Table 4. When S is low, p_m should be reduced to speed up convergence. When p is low, p_m should be increased to enhance its search. At the medium period, S often becomes high and p_m needs to become high to avoid premature convergence. At the last period, S also often becomes high and p_m should be decreased to avoid breaking the optimal solution.

Table 4 Mutation rule base

p	S		
	Low	Medium	High
low	lower	low	medium
medium	low	medium	higher
high	low	medium	higher
higher	low	low	low

3.4 Flow chart of fuzzy-based adaptive genetic algorithm

In FAGA, the fuzzy control system is employed to adjust crossover probability and mutation probability. The same selection operation as IGA is applied to keep the diversity and the best individual of each generation is saved.

Algorithmic flow of FAGA is described as follows.

(1) An initial population is produced randomly.
(2) The population diversity S is the first input variables and the algorithm stage p is the second input variables. S and p are calculated based on Eqs. (8) and (12). According to Tables 3 and 4, FLC is used to calculate p_c and p_m according to different inputs.

(3) Selection operation, crossover operation and mutation operation are carried out.

(4) If algorithm has maximum generation, it ends; else it returns to the second step.

4 SIMULATION RESULT

4.1 Test function compared with FGA1 (Test 1)

The following three benchmark functions are introduced in literature [7] and employed to demonstrate the performance of the proposed algorithm.

(1) Rosenbrock function

$$\min f_1(x_1, x_2) = 100 \times (x_1^2 - x_2)^2 + (1 - x_1)^2, \\ (-2.048 \leq x_1, x_2 \leq 2.048) \quad (13)$$

(2) Rastrigin function

$$\min f_2(x_1, x_2) = 20 + x_1^2 + x_2^2 - \\ 100(\cos 2\pi x_1 + \cos 2\pi x_2), \\ (-100 \leq x_1, x_2 \leq 100) \quad (14)$$

(3) Schaffer function

$$\max f_3(x_1, x_2) = 0.5 - \frac{(\sin \sqrt{x_1^2 + x_2^2})^2 - 0.5}{[1 + 0.001(x_1^2 + x_2^2)]^2}, \\ (-1 \leq x_1, x_2 \leq 1) \quad (15)$$

In Test 1, four kinds of genetic algorithms are

Table 5 Test 1 results

		SGA	FGA1	NFGA	FAGA
f_1	AvgBest	0.1331	0.000410	0.000290	0.0002970
	AvgError	0.1331	0.000410	0.000290	0.0002970
	StedDviation	0.2091	0.000495	0.000336	0.0003540
	AvgG	80	54	31	28
f_2	AvgBest	3.7721	0.000453	0.000054	0.0000059
	AvgError	3.7721	0.000453	0.000054	0.0000059
	StedDviation	3.3157	0.000770	0.000105	0.0000155
	AvgG	96	50	46	38
f_3	AvgBest	0.9960	0.999950	0.999997779	0.99999923
	AvgError	0.0040	0.000050	0.000002221	0.00000077
	StedDviation	0.0053	0.000041	0.000004239	0.00000245
	AvgG	94	53	6	7

employed for comparing their performances and the corresponding results are shown in Table 5 and Figs. 6, 7 and 8.

Method 1 Simple Genetic Algorithm (SGA) [7] with fixed crossover rate and mutation rate, roulette selection and single-point crossover is employed.

Method 2 Fuzzy-based Genetic Algorithm (FGA1) [7] is introduced in Section 2.2.

Method 3 NFGA, which is based on FAGA, em-

ployes the same selection strategy and best saving strategy as FAGA, and is with constant crossover rate and mutation rate.

Method 4 Fuzzy-based Adaptive Genetic Algorithm (FAGA) is proposed in Section 3.

They have the same initial parameters, *i.e.*, crossover-rate $p_c = 0.8$, mutation-rate $p_m = 0.1$, population size $N = 100$, total of generation $g_{\max} = 100$, running total max run = 100, and select parameters $h = 0.05$.

AvgBest denotes the average value of optimum, AvgError represents the difference between the average optimal value and actual optimal value, StedDviation denotes standard deviation and AvgG denotes average generation.

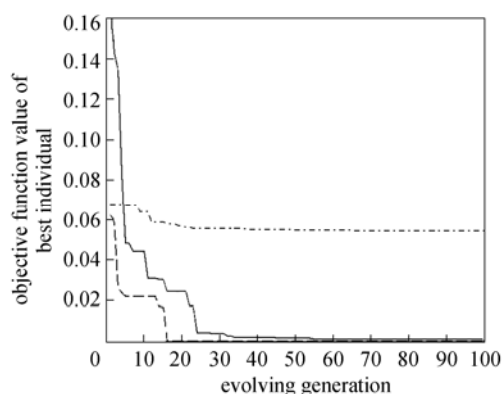


Figure 6 Convergent curve of test function 1
---- SGA; — NFGA; ---- FAGA

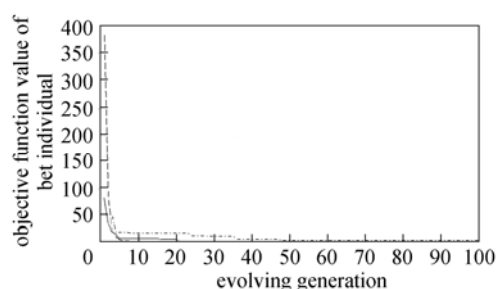


Figure 7 Convergent curve of test function 2
---- SGA; — NFGA; ---- FAGA

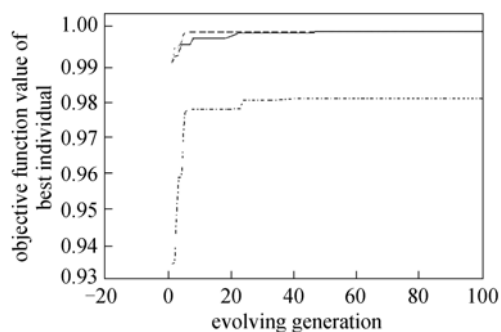


Figure 8 Convergent curve of test function 3
---- SGA; — NFGA; ---- FAGA

4.2 Test function compared with FGA2 (Test 2)

The following three benchmark functions are introduced in literature [6] and employed to demonstrate the performance of the proposed algorithm.

(1) Sphere function

$$\min f_4(x) = \sum_{i=1}^{10} x_i^2, \quad (-5.12 \leq x_i \leq 5.12) \quad (16)$$

(2) Griewangk's function

$$\min(f_5) = \frac{1}{4000} \sum_{i=1}^5 x_i^2 - \prod_{i=1}^5 \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1, \\ (-600 \leq x_i \leq 600) \quad (17)$$

(3) Needle-in-a-haystack function

$$\max(f_6) = \left[\frac{3}{0.05 + (x_1^2 + x_2^2)} \right]^2 + (x_1^2 + x_2^2)^2, \\ (-4 \leq x_1, x_2 \leq 4) \quad (18)$$

In Test 2, four kinds of genetic algorithms are employed for comparing their performances as follows. The corresponding results are shown in Table 6 and Figs. 9, 10 and 11.

Table 6 Test 2 results

		SGA	FGA2	NFGA	FAGA
f_4	AvgBest	2.5874	0.3595	1.603	0.21531
	AvgError	2.5874	0.3595	1.603	0.21531
	StedDviation	1.8801		0.790	0.10931
	AvgG	50		50	30
f_5	AvgBest	2.222	0.3525	0.728	0.26145
	AvgError	2.222	0.3525	0.728	0.26145
	StedDviation	1.167		0.289	0.15014
	AvgG	50		49	28
f_6	AvgBest	1790.84	2343.3	3597	3599.69
	AvgError	1809.16	1256.7	3	0.31
	StedDviation	1.203		7.278	0.486606
	AvgG	50		39	27

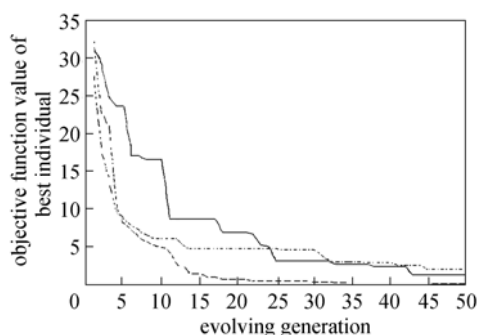


Figure 9 Convergent curve of test function 4
---- SGA; — NFGA; ---- FAGA

- Method 1 Simple Genetic Algorithm (SGA) [7].
Method 2 Fuzzy-based genetic algorithm (FGA2) [6] is introduced in Section 2.3.
Method 3 NFGA.
Method 4 Fuzzy-based Adaptive Genetic Algorithm (FAGA).

The initial parameters are the same as the literature [6], i.e., crossover-rate $p_c = 0.8$, mutation-rate $p_m = 0.1$, and select parameters $h = 0.5$.

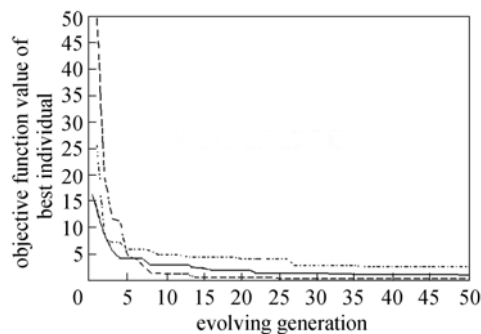


Figure 10 Convergent curve of test function 5
---- SGA; — NFGA; ---- FAGA

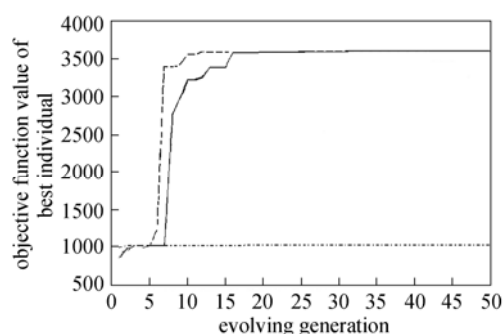


Figure 11 Convergent curve of test function 6
---- SGA; — NFGA; ---- FAGA

It can be seen from Tables 5 and 6 that the whole performance of FAGA is the best and of SGA is the worst. The performance of FAGA is better than FGA1 and FGA2. AvgError and StedDviation of FAGA are all smaller than FGA1 and FGA2, namely, FAGA has better robust and accuracy performance than FGA1 and FGA2. Considering the values of AvgG of FAGA, SGA and NFGA, FAGA has the fastest convergent speed. Because the value of AvgG was not given in the literatures [6] and [7], it was also not in the Tables 5 and 6.

Comparing the results of FAGA and NFGA, AvgBest, AvgError, StedDviation and AvgG of FAGA are better than NFGA because FAGA employs fuzzy inferences to adjust crossover probability and mutation probability dynamically during evolving process. Comparing the results of SGA and NFGA, NFGA has better performance than SGA because SGA is easy to be premature convergence and NFGA employs immune theory to improve its selection operation and overcome the premature convergence.

Figures 6 to 11 show the evolution curves of the best individual in population during the evolving process of three algorithms for 6 test functions, respectively. It can be seen that the performance of FAGA is the best and FAGA is the earliest to search into the scope of the global optimum. It is demonstrated that FAGA has faster convergent speed.

In conclusion, FAGA has good global searching ability because FAGA employs immune theory to improve its selection operation and keep the population diversity, and uses FLC to adjust p_c and p_m dynamically

Table 7 Comparison of the simulation results with the reported data

Method	k_1	E_1	k_2	E_2	k_3	E_3	EQS
Powell	0.152	62073	8.18	2384	0.221	18949	6.677
FAGA	0.2036	59553	8.19796	0.0068	0.0000019	19375.76	3.90608
NFGA	0.2129	59923	8.041	0.32	0.0000002384	16745	3.9065
SGA	0.3956	65302.3	8.8412	2381	0.062644	19588.785	3.95626

during evolving process according to the generation stage and searching performance.

5 ENGINEERING APPLICATION

Cs-Rb-V series catalyst is a newly developed sulfuric acid catalyst used under low temperature [14]. Its initial combustion temperature is 30–50 °C, being lower than common sulfuric acid catalyst. That is to say that its initial combustion temperature could be reduced to about 370 °C in comparison with normal operation temperature of 380–520 °C. Furthermore, its preparation process is simple and its preparation cost is cheaper compared with other cesium-containing low temperature catalyst. A three-step reaction mechanism for SO₂ oxidation on Cs-Rb-V low temperature sulfuric acid catalyst promoted by mixture of alkali salts was proposed. A kinetic equation was derived as follows:

$$r = \frac{K_1 p^{1/2}(\text{O}_2)}{K_2 + K_3 p(\text{SO}_3) + \frac{p(\text{SO}_3)}{p(\text{SO}_2)}} \left[1 - \frac{p(\text{SO}_3)}{p(\text{SO}_2) p^{1/2}(\text{O}_2) K_p} \right] \quad (19)$$

where $p(\text{O}_2)$, $p(\text{SO}_2)$ and $p(\text{SO}_3)$ are expressed as partial pressure of O₂, SO₂ and SO₃, respectively, r is reaction rate. K_1 , K_2 and K_3 are reaction coefficient, K_β is expressed by pre-exponential factor (k_β) and activation energy (E_β), and $K_\beta = k_\beta \exp(-E_\beta / RT)$ ($\beta = 1, 2, 3$), where R is gas constant and T is absolute temperature. K_p could be expressed by the following equation:

$$K_p = \exp \left[2.3026 \left(\frac{4812.3}{T} \times 2.8254 \lg T \right) + 2.284 \times 10^{-3} T - 7.012 \times 10^{-7} T^2 + 1.197 \times 10^{-10} T^3 + 2.23 \right] \quad (20)$$

k_β and E_β ($\beta = 1, 2, 3$) are six optimal variables, and the optimized objective function is shown as follows.

$$\min \text{EQS} = \sum_{\mu=1}^{N1} \left(\frac{r_\mu - r_{c,\mu}}{r_\mu} \right)^2 \quad (21)$$

where EQS is the sum of fitting errors, $r_{c,\mu}$ is reaction rate calculated by kinetic equation, r_μ is reaction rate of SO₂ and $N1$ is the number of experiment sample.

Algorithm estimated six kinetic parameters [k_β and E_β ($\beta = 1, 2, 3$)]. The sample of experimental data came from the literature [14]. Algorithm ran independently 10 times and each time had a maximum of 200000 evaluations. The optimal value of 10 times is shown in Table 7. The result of FAGA is 3.90608. The optimal value of FAGA is better than others and its parameter is more reasonable.

6 CONCLUSIONS

FAGA combined with fuzzy logic control and selection strategy of immune algorithm is introduced. Selection strategy of immune algorithm reduces the selection pressure and increases diversity of population. FLC is employed to dynamically adjust p_c and p_m , and the real-time and optimal parameter values for GA are obtained. It is shown in Tests 1 and 2 that the performance of the proposed method is better than FGA1 and FGA2. FAGA converges quickly and effectively overcomes the premature convergence. Finally, the satisfactory result is obtained when FAGA is used for the parameters estimation of reaction dynamics model. The optimal value obtained by the proposed algorithm is better than Powell and two other GAs.

REFERENCES

- He, H., Qian, F., "New adaptive control strategies for parameters of genetic algorithms", *Journal of East China University of Science and Technology (Natural Science Edition)*, **32** (5), 601–606 (2006). (in Chinese)
- Buchanan, B.G., Glick, J., "AI topics, a responsibility to celebrate AI responsibly", *AI Magazine*, **23** (1), 87–94 (2002).
- Yuan, X.H., Cao, L.X., Liang, Z., "Adaptive genetic algorithm with the criterion of premature convergence", *Journal of Southeast University (English Edition)*, **19** (1), 40–43 (2003).
- Tuson, A.L., Ross, P., "Adapting operator settings in genetic algorithms", *Evolutionary Computation*, **6** (2), 161–184 (1998).
- Francisco, H., Manuel, L., "Adaptive genetic operators based on co-evolution with fuzzy behaviors", *IEEE Trans. on Evolutionary Computation*, **5** (2), 149–165 (2001).
- He, Y.P., He, H., Zhang, X.L., "Evaluation index of population diversity in genetic algorithm", *Journal of Harbin Institute of Technology*, **41** (3), 67–70 (2009).
- Lan, L.K., Zhu, J.W., Liu, W.L., Yuan, Y.R., Hu, X.M., Xiao, J.,

- Zhang, J., "New method of fuzzy-based genetic algorithms", *Computer Engineering and Design*, **29** (14), 3714–3722 (2008).
- 8 Wang, K., "A new fuzzy genetic algorithm based on population diversity", In: Proceedings IEEE International Symposium on Computational Intelligence in Robotics and Automation, the Institute of Electric and Electronics Engineers, USA, 108–112 (2001).
- 9 Chang, T.M., Yih, Y.H., "Constructing a fuzzy rule system from examples", *Integrated Computer-Aided Engineering*, **6** (3), 213–222 (1999).
- 10 Freitas, A.A., Timmis, J., "Revisiting the foundations of artificial immune systems for data mining", *IEEE Transactions on Evolutionary Computation*, **11** (4), 521–540 (2007).
- 11 Wu, X.J., Zhang, Z.Z., Zhong, Y., "Genetic algorithm combined with immune mechanism and its application in skill fuzzy control", *Journal of Systems Engineering and Electronics*, **16** (3), 600–605 (2005).
- 12 Tan, G.Z., Zhou, D.M., Jiang, B., Dioubate, M.I., "Elitism-based immune genetic algorithm and its application to optimization of complex multi-modal functions", *J. Cent. South Univ. Technol.*, **15**, 845–852 (2008).
- 13 Lin, J.X., Shen, J., Xiao, G.T., Li, Y.G., Wang, P.H., "Immune genetic optimization algorithm based on multilayer fuzzy control", *Journal of Southeast University (Natural Science Edition)*, **35** (1), 46–49 (2005).
- 14 Chen, Z.X., Ye, H., Liu, J., "Study on the mechanism for the low temperature SO₂ oxidation with Cs-Rb-V sulfuric acid catalyst", *Chemical Reaction Engineering and Technology*, **17** (2), 119–124 (2001).